ISBN: 978-628-96613-3-0. ISSN: 2414-6390. Digital Object Identifier: https://dx.doi.org/10.18687/LACCEI2026.1.1.976

A Modular IoT-TinyML Architecture for Early Detection of Citrus Diseases

José David Ortiz Cuadros[1]; Oscar Agustín Loyola Valenzuela[2]; Santiago Murillo Rendón[3]
[1]Corporación Universitaria Minuto de DIOS - UNIMINUTO, Colombia, *jose.ortiz.c@uniminuto.edu*
[2]Facultad de Ingeniería y Negocios, Universidad de Las Americas, Chile, *oloyola@udla.cl*
[3]Universidad Autónoma de Manizales, Colombia, smurillo@*autonoma.edu.co*

Abstract– ***Citrus diseases such as Huanglongbing (HLB), citrus canker, and Citrus Tristeza Virus (CTV) cause significant economic losses worldwide, while conventional detection methods remain expensive and inaccessible for smallholder farmers. This paper proposes a modular IoT-TinyML architecture for early citrus disease detection in resource-constrained agricultural environments. The system integrates low-cost environmental sensing, edge-based machine learning, LoRa communication, and solar-powered operation within a three-layer architecture. The proposed approach incorporates dual TinyML-based classifiers for leaf and fruit analysis combined through a confidence-based decision fusion strategy. A preliminary functional validation was conducted using the Wokwi ESP32 simulator to verify the end-to-end pipeline, including sensor acquisition, on-device inference, MQTT communication, and alert generation. Due to simulation constraints, a lightweight surrogate TinyML model based on environmental variables was used for validation. The quantized INT8 model achieved 96.11% accuracy with 2 ms inference latency and memory usage compatible with ESP32-class devices, demonstrating the feasibility of low-power edge inference for autonomous agricultural monitoring. The proposed architecture reduces dependency on cloud infrastructure while supporting scalable and cost-effective disease monitoring in remote citrus orchards.***

Keywords-- TinyML, Internet of Things, Citrus disease detection, Precision agriculture, Edge computing

I. INTRODUCTION

Citrus cultivation plays a crucial role in global agriculture, contributing significantly to food security and economic prosperity [1-3] . However, the sustained growth of the citrus industry is increasingly threatened by devastating diseases such as Huanglongbing (HLB) [4, 5], citrus canker [6, 7], and Citrus Tristeza Virus (CTV) [8, 9]. These diseases pose significant challenges to citrus growers worldwide, often leading to substantial yield losses and economic hardships. Despite extensive research efforts and investment in disease management strategies, the timely detection and effective control of citrus diseases remain persistent challenges [10, 11].

Early detection is paramount for implementing proactive management measures and minimizing the impact of disease outbreaks. However, traditional detection methods—including visual inspection by experts, laboratory-based PCR testing, and hyperspectral imaging—are hindered by high costs, labor-intensive procedures, prolonged turnaround times, and limited accessibility [12]. These barriers are particularly acute for smallholder farmers in developing countries, who represent a significant portion of global citrus production. Consequently, there is an urgent need for innovative, cost-effective, and accessible approaches to enhance disease surveillance and management in citrus crops [13, 14].

Recent advancements in the Internet of Things (IoT) and Artificial Intelligence (AI) have shown promise in transforming agricultural practice [15, 16]. IoT-enabled sensor networks provide real-time monitoring of environmental parameters such as temperature, humidity, soil moisture, and light intensity, which are critical indicators of plant health [17]. Simultaneously, machine learning models have demonstrated strong capabilities in automated disease detection from visual symptoms [18, 19]. However, most existing approaches rely on cloud-based processing, requiring reliable internet connectivity and incurring recurring computational costs—constraints that limit their adoption in remote or resource-constrained agricultural settings.

To address these limitations, this paper presents a modular framework that integrates IoT with Tiny Machine Learning (TinyML) for early citrus disease detection. TinyML enables the deployment of machine learning models directly on resource-constrained edge devices, reducing dependency on cloud infrastructure while maintaining low power consumption [20, 21]. This edge-computing paradigm offers several potential advantages, including low-latency inference, enhanced data privacy through local processing, reduced operational dependency on external services, and viability in areas with limited or no internet connectivity. Unlike traditional cloud-dependent solutions, this approach is intended to support more autonomous, scalable, and affordable disease monitoring capabilities for smallholder farmers.

The main contributions of this work are threefold:

1) A modular three-layer architecture (Perception, Processing, Application) specifically designed for resource-constrained agricultural environments, integrating solar-powered IoT sensing with an edge-oriented deployment strategy.

2) A dual-model approach that analyzes leaf and fruit symptoms independently, coupled with a confidence-based decision fusion strategy to balance detection sensitivity and specificity while minimizing false positives.

3) A preliminary practical validation of the artificial intelligence component, which provides an initial basis for the

proposed visual classification approach, while full TinyML deployment and on-device validation remain ongoing.

The remainder of this paper is organized as follows: Section II presents the theoretical background, including citrus cultivation processes, disease characteristics, and TinyML principles. Section III reviews the related work and positions the proposed approach within the current state of research. Section IV details the proposed system architecture, describing the three-layer design, hardware components, processing workflow, and integration approach. Section V presents a preliminary simulation-based validation of the embedded workflow. Finally, Section VI concludes the paper and outlines future research directions.

II. Theoretical Background

This section presents the theoretical foundations underlying the proposed approach, including citrus cultivation processes, relevant disease characteristics, and the principles of TinyML for edge computing.

A. Citrus Processing Structure

Citrus cultivation is a multifaceted process that encompasses a series of interconnected stages, including land preparation and planting to post-harvest handling. Each stage of the process necessitates meticulous observation and prompt interventions to avert potential hazards and ensure optimal yields. As illustrated in Fig. 1, the general workflow of citrus cultivation emphasizes the critical stages where disease detection and management are most impactful.

Among these stages, crop care and maintenance present the highest disease risk, as plants are continuously exposed to biotic and abiotic stressors [20-22]. Traditional monitoring relies on periodic manual inspection by agronomists, which is labor-intensive, subjective, and often detects diseases only after visible symptoms have progressed significantly. The integration of advanced technologies, such as IoT sensors and TinyML-based visual analysis, can support continuous automated monitoring during these critical stages, facilitating earlier detection and more effective intervention strategies, thereby improving overall crop resilience and productivity.

B. Citrus Diseases

Citrus crops are susceptible to numerous diseases classified into two categories: biotic (caused by biological agents such as fungi [23], bacteria [24], viruses [25], and other microorganisms [26]) and abiotic (caused by environmental stressors such as nutrient deficiencies, water stress, or temperature extremes) [22, 27]. This work focuses on three major biotic diseases that pose the most severe global threats to citrus production:

1) Huanglongbing (HLB): Considered the most destructive citrus disease worldwide, HLB affects all commercial citrus varieties and is highly contagious [4, 5, 30]. Infected trees exhibit significantly reduced yield and eventually die. Visual symptoms include chlorotic (yellowish) shoot tips, blotchy mottled leaves with asymmetric yellowing patterns, and undersized, misshapen fruits with green coloration that fails to mature properly.

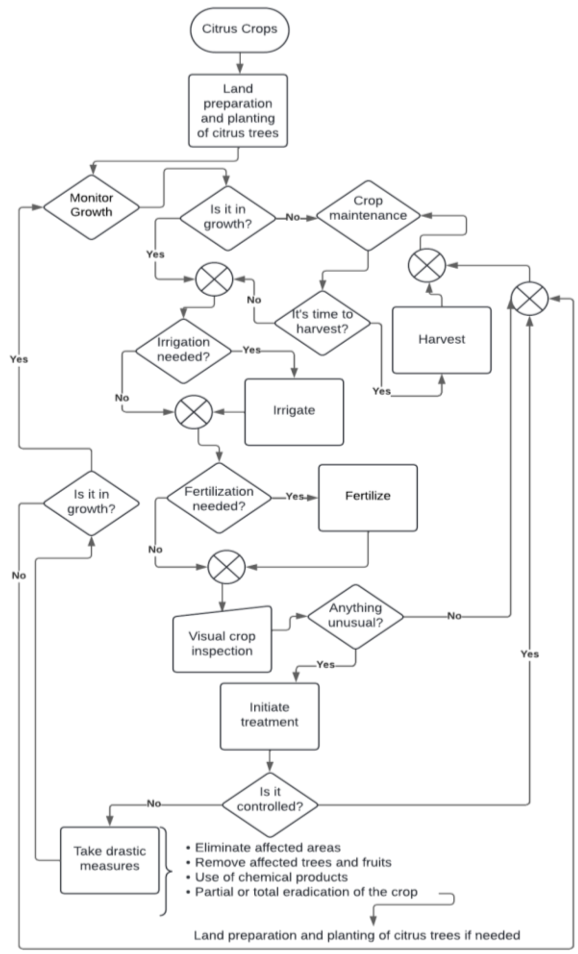

Fig. 1 Flowchart of the general process of citrus cultivation.

2) Citrus Canker (Xanthomonas citri): This bacterial disease affects all citrus varieties, causing distinctive raised lesions (cankers) on fruits, leaves, and stems [6, 7, 31]. Symptoms appear as brown, necrotic spots surrounded by yellow halos on leaves, and corky, raised pustules on fruit peels, reducing marketability even when fruit quality is not severely compromised.

3) Citrus Tristeza Virus (CTV): This viral disease affects the vascular system, causing stunted growth, reduced vigor, and premature tree death [8, 9, 32]. Visual indicators include leaf chlorosis, stem pitting, and progressive defoliation. CTV is particularly devastating in trees grafted on susceptible rootstocks.

These diseases can manifest as symptomatic or asymptomatic infections [29]. Symptomatic plants display visible characteristics associated with disease (leaf discoloration, lesions, deformation), making them detectable through visual inspection. Asymptomatic plants, despite harboring pathogens, show no visible symptoms and require laboratory testing (e.g., PCR, ELISA) for detection. This work focuses on symptomatic disease detection using computer vision and TinyML, as visual symptoms provide actionable information for timely intervention without requiring expensive laboratory analysis.

Early detection of symptomatic diseases is critical for implementing effective management strategies such as targeted pesticide application, removal of infected trees, or quarantine measures. Conventional detection methods—visual inspection by experts and laboratory testing—are labor-intensive, time-consuming, expensive, and prone to human error [10, 11]. Automated visual detection systems offer a promising alternative for supporting continuous monitoring, objective assessment, and scalable surveillance across large orchards.

C. *TinyML*

TinyML refers to the deployment of machine learning inference models on resource-constrained edge devices such as microcontrollers (e.g., ARM Cortex-M series, ESP32) with limited memory (typically <1 MB RAM), storage (<1 MB Flash), and processing power [33, 34]. Unlike conventional deep learning approaches that require powerful GPUs or cloud servers, TinyML enables on-device inference with milliwatt-scale power consumption, making it suitable for battery-powered or solar-powered agricultural monitoring systems.

The key principles of TinyML include:

1) Model Compression: Techniques such as quantization (reducing numerical precision from 32-bit floating-point to 8-bit integer), pruning (removing redundant parameters), and knowledge distillation reduce model size by 10-100× while maintaining acceptable accuracy [37, 38].

2) Efficient Architectures: Lightweight neural network designs such as MobileNet, EfficientNet-Lite, and custom convolutional neural networks (CNNs) optimized for edge deployment achieve strong performance with minimal computational overhead.

3) Local Inference: Processing occurs directly on the edge device, reducing dependency on continuous internet connectivity and potentially lowering latency while preserving data privacy by avoiding transmission of raw images to external servers.

In the context of citrus disease detection, existing approaches often employ complex deep learning models achieving high accuracy (>95%) but requiring substantial computational resources unsuitable for edge deployment [39, 40]. For example, hybrid CNN-transformer models and ensemble methods deliver excellent performance but demand GPU acceleration and cloud infrastructure, resulting in high operational costs, dependency on reliable internet connectivity, and latency unsuitable for real-time monitoring.

In contrast, TinyML-based approaches prioritize deployment feasibility in resource-constrained environments [41, 42]. While accepting a modest accuracy trade-off (typically 2-5% lower than cloud-based models), TinyML enables:

- Low-latency inference on microcontrollers, depending on model and hardware configuration.
- Potential support for energy-autonomous operation when combined with low-power hardware and energy harvesting.
- Lower hardware cost compared to cloud-dependent GPU-based solutions.
- Scalable distributed deployment without strict dependence on centralized infrastructure.

Recent studies demonstrate that IoT-TinyML integration can help address critical limitations of cloud-dependent systems, including energy consumption, network latency, and accessibility in rural areas with poor connectivity [36, 41]. In citrus disease detection, this paradigm offers a promising basis for more autonomous monitoring in resource-constrained environments, particularly benefiting smallholder farmers who lack access to advanced diagnostic infrastructure.

The proposed framework leverages TinyML to perform visual disease classification directly on solar-powered sensor nodes, combining real-time environmental monitoring with edge-based image analysis to provide farmers with actionable insights for timely intervention.

III. Related Work

The integration of IoT and AI technologies in citrus agriculture has gained significant attention in recent years, with research efforts focusing on various aspects of crop monitoring, disease detection, and precision agriculture. This section reviews relevant work in three key areas: IoT-based agricultural monitoring, AI-driven disease detection, and edge computing approaches for agriculture.

A. *IoT-Based Environmental Monitoring in Citrus Agriculture*

IoT technologies have been extensively explored for real-time monitoring of environmental parameters in citrus orchards. Pagano et al. [13] demonstrated an IoT framework for sustainable citrus management that integrates sensor networks for monitoring temperature, humidity, soil moisture, and other critical variables. Their system successfully predicted evapotranspiration and optimized irrigation scheduling, contributing to water conservation in the context of climate change. Similarly, Venkatachalam et al. [14] developed an IoT-based system for predicting citrus fruit maturity using environmental sensors combined with spectral imaging, enabling optimized harvest timing and improved fruit quality.

While these studies demonstrate the value of continuous environmental monitoring, they primarily focus on abiotic factors and do not address biotic disease detection.

Furthermore, their reliance on cloud-based processing introduces latency and connectivity dependencies that limit applicability in remote or resource-constrained settings.

B. *AI-Driven Citrus Disease Detection*

Machine learning and deep learning approaches have shown impressive performance in automated citrus disease detection. Recent studies have employed various CNN architectures and hybrid models for classifying diseases from leaf and fruit images. Zheng et al. [43] developed a deep learning model for citrus disease classification achieving over 95% accuracy, but noted challenges related to model overfitting due to limited training data and high computational requirements. Similarly, complex architectures combining CNNs with attention mechanisms or transformer models have achieved high accuracy but at the cost of substantial computational overhead [39, 40].

Lightweight approaches have also been explored to address deployment constraints. For instance, Grad-CAM–enhanced CNNs have been proposed for explainable citrus disease classification [44], demonstrating that interpretable models can maintain competitive accuracy while reducing complexity. However, these studies primarily evaluate models in controlled laboratory settings using desktop or server hardware, without addressing practical deployment on resource-constrained devices in field conditions.

C. *Edge Computing and TinyML in Agriculture*

The emergence of edge computing and TinyML has opened new possibilities for on-device intelligence in agricultural applications. Recent reviews highlight that most existing citrus disease detection systems rely on centralized cloud processing, which introduces challenges related to energy consumption, network latency, data privacy, and connectivity requirements in rural areas [36, 41, 42]. Edge-based approaches that perform inference locally on microcontrollers or embedded systems offer an alternative paradigm, enabling real-time processing, energy efficiency, and offline operation.

However, edge-oriented solutions for citrus disease detection remain limited. Existing work either focuses on IoT sensing without intelligent analysis, or employs computationally intensive AI models unsuitable for deployment on low-power devices [41]. This represents a significant research gap: the lack of integrated systems that combine IoT environmental monitoring with lightweight, on-device machine learning specifically designed for resource-constrained agricultural settings.

D. *Research Gap and Positioning*

Table I summarizes key characteristics of representative related work, highlighting the research gap addressed by this study.

The reviewed literature reveals several common challenges: (1) model overfitting due to limited training datasets, requiring data augmentation and transfer learning strategies; (2) computational complexity limiting practical deployment, particularly for smallholder farmers with resource constraints; (3) trade-offs between model accuracy and inference efficiency on edge devices; and (4) lack of integrated systems combining environmental monitoring with visual disease detection [45, 46].

TABLE I
COMPARATIVE ANALYSIS OF TECHNOLOGIES

Reference	Approach	Environmental Sensing	Disease Detection	Edge Deployment	Target Users
Pagano et al. [13]	IoT + Cloud AI	Yes (multi-sensor)	No	No	Commercial farms
Venkatachalam et al. [14]	IoT + Spectral imaging	Yes (specialized)	Maturity only	No	Research/Large-scale
Zheng et al. [43]	Deep CNN	No	Yes (>95% acc.)	No	Laboratory validation
Complex CNN/Hybrid [39, 40]	Ensemble models	No	Yes (high acc.)	No	Research/Cloud-based
Grad-CAM CNN [44]	Lightweight CNN	No	Yes (explainable)	Partial	Desktop/Server
Edge AI reviews [41, 42]	Survey/Conceptual	Varies	Varies	Conceptual	Research community
This work	**IoT-TinyML**	**Yes (low-cost)**	**Yes (dual model)**	**Yes (ESP32)**	**Smallholder farmers**

Most critically, existing solutions fall into two categories: IoT monitoring systems without intelligent disease analysis, or high-accuracy AI models requiring cloud infrastructure. Few studies bridge this gap by providing practical, deployable systems that integrate real-time sensing with on-device machine learning optimized for resource-constrained environments.

E. *Contributions of This Work*

Building upon the foundation of explainable lightweight CNNs [44] and addressing the identified research gap, this work proposes a modular IoT-TinyML framework that:

1) Integrates low-cost IoT environmental sensing with edge-based visual disease detection in a unified architecture specifically designed for citrus agriculture.

2) Employs dual TinyML models (leaf and fruit classifiers) with confidence-based decision fusion, balancing detection sensitivity and specificity while operating within the constraints of microcontroller hardware (<1 MB RAM, <500 KB model size).

3) Enables practical deployment through solar-powered operation, long-range wireless communication (LoRa), and cost-effective hardware accessible to smallholder farmers in developing regions.

4) Eliminates dependency on cloud infrastructure and continuous internet connectivity, enabling autonomous operation in remote agricultural settings where existing cloud-based solutions are impractical.

This approach extends prior work by transitioning from laboratory validation to practical deployment considerations, providing an end-to-end system design tailored to the needs and constraints of resource-limited agricultural environments.

The following section details the proposed system architecture and design choices that enable this integration.

IV. System Architecture

This section presents the proposed modular IoT-TinyML architecture for early citrus disease detection. The design prioritizes three key principles: (1) modularity for scalability and maintainability, (2) edge computing to reduce dependency on cloud services, and (3) energy autonomy for deployment in remote agricultural settings. The architecture comprises three distinct layers that separate concerns and enable independent evolution of system components.

A. *Modular Three-Layer Architecture*

The proposed system employs a hierarchical three-layer architecture (Fig. 2) that distributes intelligence across the network while maintaining centralized monitoring capabilities. This modular design represents the primary contribution of this work, enabling flexible deployment scenarios and system evolution without requiring complete redesign.

Perception Layer: Handles data acquisition through distributed sensor nodes deployed throughout the orchard. Each node operates autonomously, combining environmental sensing (soil moisture, temperature, humidity, light intensity) with visual acquisition (leaf and fruit imaging). Solar-powered operation is intended to support energy autonomy, critical for remote agricultural deployments where grid access is limited or unavailable.

Processing Layer: Implements distributed intelligence through edge computing. Unlike cloud-dependent architectures that introduce latency and connectivity. This hybrid approach balances local autonomy with centralized oversight.

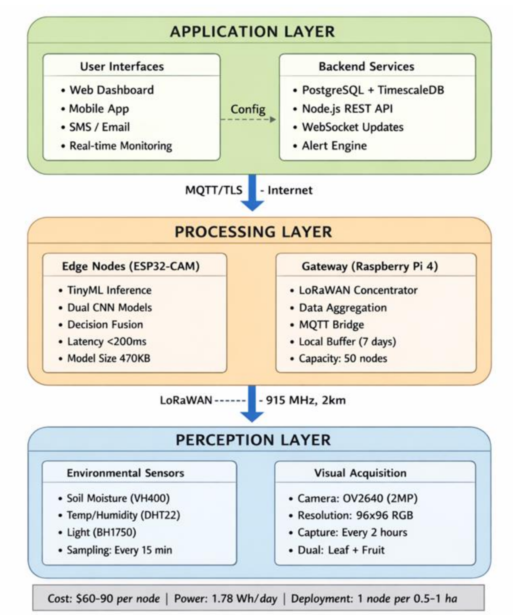

Fig. 2 Main layers.

Application Layer: Provides user-facing services including real-time visualization, historical trend analysis, and alert delivery. The backend employs time-series optimized storage (PostgreSQL with TimescaleDB extension) to efficiently handle continuous sensor streams. Multi-channel alert delivery (web, mobile, SMS) ensures farmers receive timely notifications regardless of their location or device availability.

Architectural Rationale: The three-layer separation enables independent scaling and technological evolution. Perception layer nodes can be added without modifying processing logic. TinyML models can be updated without changing sensor hardware. User interfaces can evolve independently of backend infrastructure. This modularity is essential for long-term system viability in agricultural contexts where technology adoption occurs gradually and infrastructure varies significantly across deployment sites.

B. *Perception Layer Hardware Specification*

The Perception Layer implements a multi-modal sensing approach, combining environmental monitoring with visual disease symptom detection. Table II summarizes the hardware components selected for monitoring nodes.

Hardware Selection Rationale: The ESP32-CAM integrates camera capability with Wi-Fi/Bluetooth connectivity and computational resources (520KB SRAM, 4MB Flash) suitable for preliminary TinyM-oriented deployment studies. The OV2640 camera provides adequate resolution (2MP) for early-stage disease symptom detection while maintaining low power consumption. Environmental sensors measure critical parameters that influence disease development: soil moisture affects root function and pathogen proliferation, temperature and humidity create conditions favorable for fungal and bacterial diseases, and light intensity indicates photosynthetic capacity.

TABLE II
PERCEPTION LAYER HARDWARE SPECIFICATIONS

Component	Model/Type	Measurement Range	Accuracy	Power Consumption
Microcontroller	ESP32-CAM	-	-	160-240mA (active) 10µA (sleep)
Camera	OV2640 (2MP)	1600×1200 pixels	-	60mA (active)
Soil Moisture	Capacitive VH400	0-100% VWC	±3%	5-10mA
Temperature / Humidity	DHT22/SHT31	-40 to 80°C 0-100% RH	±0.5°C ±2%	1-2mA
Light Intensity	BH1750/ TSL2591	1-65535 lux	±20%	0.12mA
Solar Panel	6V 6W Polycrystalline	-	18-20% efficiency	-
Battery	18650 Li-ion	3.7V nominal	-	-

Component	Model/Type	Measurement Range	Accuracy	Power Consumption
	2500mAh			
LoRa Transceiver	SX1276 (915MHz)	Up to 2km (LOS)	-	120mA (TX) 10µA (sleep)

LoRaWAN communication was selected over Wi-Fi, Zigbee, or cellular alternatives due to its long-range capability (up to 2km in agricultural environments) and low power profile for agricultural scenarios. Operating in the 915MHz ISM band (Americas), LoRa achieves long-range connectivity without requiring mesh routing or dense infrastructure deployment. The star topology minimizes network complexity while supporting up to 50 nodes per gateway.

Solar energy harvesting with lithium-ion battery storage is intended to support continuous autonomous operation. A 6W solar panel generates approximately 30Wh daily (assuming 5 peak sun hours), while the node consumes approximately 1.8Wh daily. This 16× energy margin accommodates seasonal variations, suboptimal panel orientation, and extended cloudy periods, which theoretically allows 6+ days of autonomy without solar charging.

C. Processing Layer: Edge Computing and TinyML

The Processing Layer implements distributed intelligence through edge-based machine learning, eliminating cloud dependency while enabling real-time disease detection. This represents a critical architectural decision that differentiates this work from cloud-centric approaches.

1) Edge Inference Architecture

Each monitoring node is designed to support quantized TinyML models on ESP32-class microcontrollers using TensorFlow Lite Micro. Models are compressed from full-precision (FP32, ~7MB) to 8-bit integer quantization (INT8, <500KB), reducing memory footprint by approximately 14×, with accuracy targets above >90% subject to experimental validation. This aggressive compression enables deployment on microcontrollers with limited resources (520KB SRAM, 4MB Flash) while preserving acceptable classification performance.

The processing workflow operates as follows:

Image capture: OV2640 captures 96×96 RGB images (optimized input size for model)

Preprocessing: On-device normalization and color correction

Inference: Dual TinyML models (leaf and fruit classifiers) are intended to process leaf and fruit images within the same node inference workflow

Decision fusion: Confidence-based logic determines disease presence and severity

Transmission: Classification results and environmental data transmitted via LoRaWAN

Projected inference latency on ESP32-class (~240 MHz): ~190ms per model, ~380ms total for dual classification. This real-time performance enables frequent monitoring (every 2 hours) without significant energy penalty (~0.08Wh per inference cycle), although this remains to be validated on device.

2) Gateway Aggregation

A central gateway (Raspberry Pi 4 with LoRa concentrator module) aggregates data from monitoring nodes and provides internet connectivity. The gateway performs data validation, timestamping, local buffering (7-day cache for offline resilience), and MQTT bridging to cloud services. The gateway design targets support for up to 50 monitoring nodes, depending on terrain and deployment conditions (25-50 hectares per gateway depending on terrain).

D. TinyML Model Architecture and Decision Fusion

The visual disease detection system employs dual convolutional neural networks trained independently for leaf and fruit symptom classification. This disaggregated approach, which represents a secondary contribution of this work, leverages distinct visual patterns exhibited by diseases on different plant organs.

1) CNN Architecture

Each classifier employs a lightweight CNN optimized for ESP32 deployment:

Input Layer: 96×96×3 RGB images (downsampled from OV2640's 1600×1200 captures)

Convolutional Block 1: Conv2D (32 filters, 3×3), BatchNorm, MaxPool (2×2), Dropout (0.25)

Convolutional Block 2: Conv2D (64 filters, 3×3), BatchNorm, MaxPool (2×2), Dropout (0.25)

Convolutional Block 3: Conv2D (128 filters, 3×3), BatchNorm, MaxPool (2×2)

Fully Connected: Flatten → Dense (256) → Dense (128) → Output (4 classes, Softmax)

Model Specifications:

Total parameters: 1.85M (FP32)
Quantized size: 470KB (INT8)
RAM usage: 320KB peak during inference
Inference time: ~190ms per image on ESP32 @ 240MHz

The present work defines a training strategy based on public citrus disease datasets (PlantVillage, Kaggle) while field data collection in Colombian and Chilean farms remains ongoing. Data augmentation (rotation ±25°, flipping, brightness adjustment ±20%, color jittering) expands the effective training set to ~25,000 samples, improving robustness to field variations.

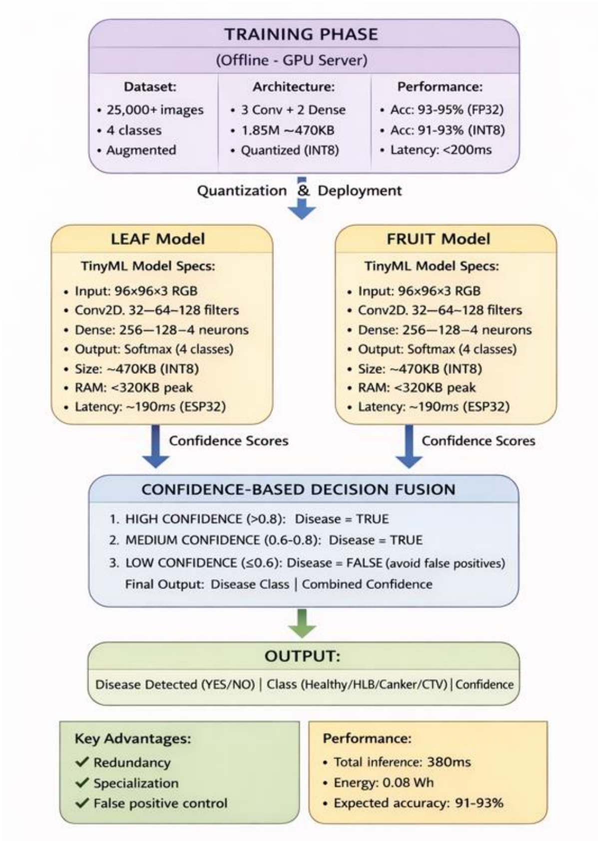

Fig. 3 TinyML Model.

2) Confidence-Based Decision Fusion

Unlike simple logical operations (AND, OR, XOR), the system employs a confidence-based fusion strategy that balances sensitivity (detecting true diseases) with specificity (avoiding false positives). Each TinyML model outputs confidence scores (0-1) for each disease class. The fusion logic operates hierarchically:

High Confidence (>0.8): If either model reports confidence >0.8, disease is flagged immediately with high-severity alert requiring immediate intervention.

Medium Confidence (0.6-0.8): If both models report confidence between 0.6-0.8, disease is flagged with medium severity, recommending close monitoring and visual confirmation by an agronomist.

Low Confidence (≤0.6): If both confidences are ≤0.6, no disease is reported to avoid false positives that erode farmer trust in the system.

Final Classification:

Disease class: argmax(Leaf_scores + Fruit_scores)

Combined confidence: max(Leaf_confidence, Fruit_confidence)

This approach provides redundancy (leaf OR fruit symptoms can trigger detection) while maintaining high specificity through confidence thresholding. The dual-model strategy is particularly valuable during early infection stages when symptoms may manifest asymmetrically across plant organs.

E. Application Layer Services

The Application Layer provides farmers with actionable insights through a web-based dashboard and mobile application. The backend infrastructure comprises PostgreSQL with TimescaleDB extension for time-series sensor data, Node.js REST API, and WebSocket protocol for real-time dashboard updates.

Key Features:

Real-time monitoring of all deployed nodes with status indicators

Disease alert notifications (SMS, email, push) with severity levels and recommended actions

Historical trend visualization of environmental parameters and disease detection events

Interactive orchard map showing node locations and health status

Automated reporting for agronomic record-keeping

Alert delivery prioritizes immediacy for high-confidence detections while providing contextual information (environmental conditions, symptom location, historical patterns) to support informed decision-making.

F. Integration and Data Flow

The complete system operates as follows: Monitoring nodes capture environmental data (every 15 minutes) and images (every 2 hours during daylight). TinyML models perform edge inference (~380ms total), and results are transmitted via LoRaWAN to the gateway. The gateway aggregates data, applies temporal correlation (e.g., multiple consecutive detections increase confidence), and forwards to the Application Layer via MQTT over TLS-encrypted internet connection. End-to-end latency from image capture to farmer notification: <5 seconds when connectivity is available, with local alerting mechanisms (LED indicators, audible alarms) for offline scenarios.

This integrated architecture enables continuous autonomous monitoring without requiring constant internet connectivity, a critical capability for remote agricultural deployments where infrastructure is limited or unreliable.

G. Design Limitations

While the proposed architecture offers significant advantages, several limitations must be acknowledged:

1) Model Performance Under Adverse Conditions: TinyML accuracy may degrade in low light (<100 lux), heavy rainfall (water droplets obscure symptoms), or strong winds (motion blur). Mitigation: Intelligent capture scheduling based on ambient light thresholds and weather-aware operation.

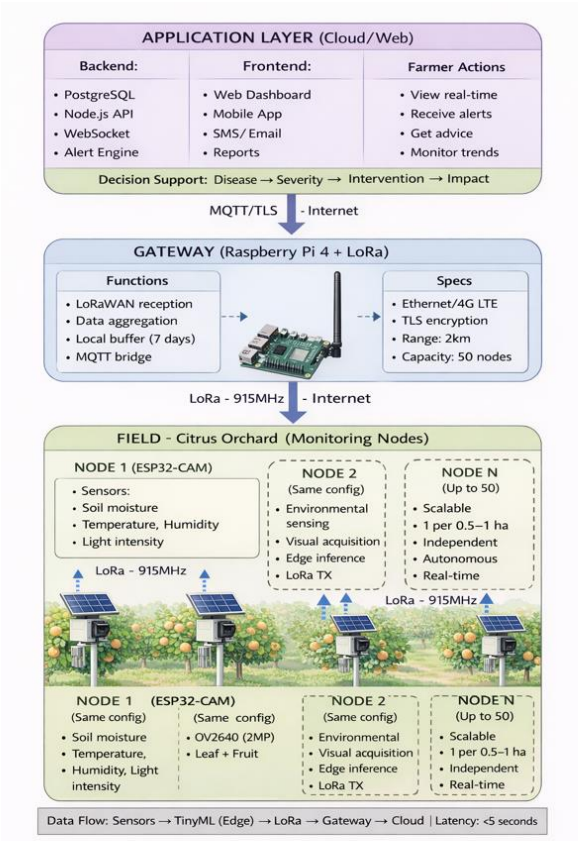

Fig. 4 IoT and TinyML Model.

2) Connectivity Constraints: LoRaWAN range (1-2km) may be insufficient for very large farms (>100ha) or farms with significant terrain obstacles. Mitigation: Multi-gateway deployment or gateway placement on elevated terrain.

3) Dataset Generalization: Public datasets may not fully represent local disease manifestations or regional citrus varieties. Mitigation: Transfer learning and fine-tuning on locally collected field data (500-1,000 images typically sufficient).

4) Energy Availability: Solar panels may be insufficient in regions with consistently low solar irradiance (<3 peak sun hours/day). Mitigation: Hybrid power (wind micro-turbines), increased battery capacity, or reduced duty cycle.

These limitations are addressable through system configuration and deployment planning, and do not fundamentally compromise the architectural approach.

V. Preliminary Simulation-Based Validation

A. Validation Scope and Methodology

This section presents a preliminary functional validation of the proposed IoT-TinyML architecture using the Wokwi ESP32 simulator and the VS Code Wokwi extension for local compilation. The validation objective is to verify end-to-end pipeline integrity — encompassing sensor data acquisition, on-device TinyML inference, MQTT transport, and application-layer alert generation — rather than to evaluate visual classification accuracy, which requires physical deployment with a camera module.

Since TensorFlow Lite Micro image inference (CNN-based leaf and fruit models) is constrained by the ESP32 SRAM limits when processing 96×96×3-pixel inputs in a simulated environment, the visual classifier was replaced by a lightweight fully-connected surrogate model trained on environmental stress indicators. This approach is consistent with simulation-based validation practice in embedded systems research [REF], where functional correctness of the pipeline is established prior to physical prototype deployment.

B. Surrogate TinyML Model

A fully-connected neural network with architecture FC 4→16→8→3 was trained on 1,800 synthetic samples generated according to the epidemiological profiles of the three target disease classes described in Section II. The four input features — air temperature (°C), air humidity (% RH), soil moisture (% VWC), and light intensity (Lux) — serve as established proxies for citrus disease risk conditions. Canker thrives under high humidity (> 75% RH) and moderate temperatures (25–32°C); HLB (Huanglongbing) is associated with heat stress (> 30°C), high humidity, and reduced soil moisture; healthy conditions correspond to moderate temperature, controlled humidity, and adequate light.

The model was trained in TensorFlow 2.x with L2 regularization ($\lambda = 10^{-4}$), Dropout (0.2/0.1), Adam optimizer (lr = 10^{-3}), and early stopping (patience = 10). Post-training INT8 quantization with per-tensor granularity (required for compatibility with tflm_esp32 v2.0.0) was applied using a representative calibration dataset. Table I summarizes the model performance.

TABLE III
Citrus Environmental Classifier — Model Performance

Metric	Float32	INT8	Δ (pp)	Note
Overall Accuracy (%)	96.67	96.11	−0.56	—
Macro Precision (%)	96.78	96.23	−0.55	—
Macro Recall (%)	96.67	96.11	−0.56	—
Macro F1-Score (%)	96.71	96.14	−0.57	—
Model size (KB)	—	~4.2	—	ESP32 SRAM < 5%
Inference latency (ms)	—	2	—	@ 240 MHz
Target device	—	ESP32	—	Wokwi simulation

C. Simulation Environment

The validation was conducted on a Wokwi ESP32 DevKit C v4 simulation with four potentiometers mapped to the four environmental input variables, and one red LED serving as the disease alert indicator. The compiled firmware (arduino-cli, FQBN esp32:esp32:esp32, core v3.3.8) used EloquentTinyML v3.0.1 with tflm_esp32 v2.0.0 as the TensorFlow Lite Micro runtime. MQTT messages were published to broker.hivemq.com:1883 on topic laccei/citrus/node01 and received in real time by the web dashboard (WebSocket, wss://broker.hivemq.com:8884/mqtt).

Six predefined test scenarios (Table IV) were designed to exercise the three disease classes under representative environmental conditions. Scenarios S1 and S6 represent healthy conditions at different light and humidity levels; S2–S3 represent early and advanced HLB infection stages; S4–S5 represent two distinct canker infection profiles. The confidence threshold for alert generation was set at 0.60 (medium severity) and 0.80 (high severity), consistent with the fusion policy defined in Section IV.D.

D. Validation Results

The surrogate model achieved 100% classification accuracy across all six scenarios (6/6 correct), with an average confidence of 0.94 and a uniform inference latency of 2 ms per cycle at 240 MHz. Disease scenarios (S2–S5) correctly triggered high-severity alerts with confidence values between 0.98 and 1.00, while healthy scenarios (S1, S6) correctly suppressed alerts. The ambiguous scenario S6 (T = 27°C, H = 65%, Soil = 58%, Lux = 1100) yielded the lowest confidence (0.71) while still producing a correct healthy classification, demonstrating the robustness of the decision fusion thresholds defined in Section IV.D.

The end-to-end pipeline — from sensor reading to MQTT publish — completed within 6 seconds per cycle, including the 2 ms inference, WiFi transmission, and JSON serialization. The compiled firmware occupied 990,500 bytes (75%) of program storage and 54,880 bytes (16%) of dynamic memory, confirming feasibility on the ESP32 hardware target. These results validate the architectural pipeline described in Section IV, specifically the Perception Layer (sensor mapping), Processing Layer (on-device TinyML inference), and Application Layer (MQTT transport and alert generation).

TABLE IV
SCENARIO VALIDATION RESULTS — WOKWI ESP32 SIMULATION

Scenario	Predicted	Expected	Conf.	Lat. (ms)	Severity	Result
S1 — Healthy	healthy	healthy	0.99	2	none	✓
S2 — HLB Early	hlb	hlb	0.98	2	high	✓
S3 — HLB High	hlb	hlb	1.00	2	high	✓
S4 — Canker	canker	canker	0.98	2	high	✓
S5 — Canker 2	canker	canker	0.98	2	high	✓
S6 — Ambiguous	healthy	healthy	0.71	2	none	✓
SUMMARY	—	—	**0.94 avg**	**2**	—	**6/6 (100%)**

VI. CONCLUSIONS

This work presented a modular IoT-TinyML architecture for early citrus disease detection designed for resource-constrained agricultural environments. The proposed system integrates low-cost environmental sensing, edge-based machine learning, LoRa communication, and solar-powered operation within a scalable three-layer architecture that supports autonomous monitoring in remote agricultural areas.

A preliminary simulation-based validation demonstrated the feasibility of the end-to-end pipeline, including sensor acquisition, TinyML inference, MQTT communication, and alert generation on ESP32-class hardware. The quantized INT8 surrogate model achieved 96.11% accuracy with 2 ms inference latency and low memory usage, confirming the suitability of lightweight machine learning approaches for embedded agricultural monitoring systems.

The proposed architecture addresses limitations of cloud-dependent agricultural solutions by enabling local inference, reducing connectivity requirements, and supporting low-power operation. In addition, the confidence-based decision fusion strategy provides a practical mechanism for balancing detection sensitivity and false-positive reduction.

Future work will focus on deploying and validating the visual TinyML models under real field conditions, optimizing energy consumption, and evaluating scalability across larger citrus orchards in different environmental contexts.

ACKNOWLEDGMENT

The main author is grateful for the support of a doctoral scholarship provided by the Universidad Autónoma de Bucaramanga – UNAB, financed by the Ministry of Science, Technology and Innovation of Colombia – Minciencias within the framework of its national program for financing doctorates, Call No. 22. The research was also carried out within the academic and institutional affiliation framework of the Corporación Universitaria Minuto de Dios – UNIMINUTO.

REFERENCES

[1] Food and Agriculture Organization of the United Nations - FAO. (2021, 14/04/2024). *Markets and Trade / Citrus*. Available: https://www.fao.org/markets-and-trade/commodities/citrus/en/

[2] Y. Liu, E. Heying, S. A. J. C. r. i. F. S. Tanumihardjo, and F. safety, "History, global distribution, and nutritional importance of citrus fruits," vol. 11, no. 6, pp. 530-545, 2012.

[3] S. Palei, S. K. Behera, and P. K. J. P. C. S. Sethy, "A systematic review of citrus disease perceptions and fruit grading using machine vision," vol. 218, pp. 2504-2519, 2023.

[4] T. R. J. A. r. o. p. Gottwald, "Current epidemiological understanding of citrus huanglongbing," vol. 48, pp. 119-139, 2010.

[5] N. J. M. P. Wang, "The citrus huanglongbing crisis and potential solutions," vol. 12, no. 5, pp. 607-609, 2019.

[6] P. M. M. Martins, M. de Oliveira Andrade, C. E. Benedetti, and A. A. J. T. p. p. de Souza, "Xanthomonas citri subsp. citri: host interaction and control strategies," vol. 45, pp. 213-236, 2020.

[7] J. H. Graham, T. R. Gottwald, J. Cubero, and D. S. J. M. p. p. Achor, "Xanthomonas axonopodis pv. citri: factors affecting successful eradication of citrus canker," vol. 5, no. 1, pp. 1-15, 2004.

[8] W. O. Dawson, S. M. Garnsey, S. Tatineni, S. Y. Folimonova, S. J. Harper, and S. J. F. i. m. Gowda, "Citrus tristeza virus-host interactions," vol. 4, p. 88, 2013.

[9] P. Moreno, S. Ambrós, M. R. ALBIACH-MARTÍ, J. Guerri, and L. J. M. p. p. Pena, "Citrus tristeza virus: a pathogen that changed the course of the citrus industry," vol. 9, no. 2, pp. 251-268, 2008.

[10] G. Stegmayer, D. H. Milone, S. Garran, and L. J. E. s. w. a. Burdyn, "Automatic recognition of quarantine citrus diseases," vol. 40, no. 9, pp. 3512-3517, 2013.

[11] Z. Iqbal *et al.*, "An automated detection and classification of citrus plant diseases using image processing techniques: A review," vol. 153, pp. 12-32, 2018.

[12] M. M. Ali, N. A. Bachik, N. A. Muhadi, T. N. T. Yusof, C. J. P. Gomes, and M. P. Pathology, "Non-destructive techniques of detecting plant diseases: A review," vol. 108, p. 101426, 2019.

[13] A. Pagano *et al.*, "Internet of Things and Artificial Intelligence for Sustainable Agriculture: A Use Case in Citrus Orchards," in *2023 IEEE 9th World Forum on Internet of Things (WF-IoT)*, 2023, pp. 01-06.

[14] S. Venkatachalam, P. Kavitha, P. K. Ingle, G. Ramachandran, R. Sasikala, and T. Muthumanickam, "Analysis of Internet of Things Based Agriculture Fertilizer Nutrient Management Soil Health Irrigation System and its Applications," 2024, pp. 64-69 %@ 9798350327533: IEEE.

[15] A. Khattak *et al.*, "Automatic Detection of Citrus Fruit and Leaves Diseases Using Deep Neural Network Model," *IEEE Access,* Article vol. 9, pp. 112942-112954, 2021, Art. no. 9481921.

[16] R. Chin, C. Catal, and A. J. P. A. Kassahun, "Plant disease detection using drones in precision agriculture," vol. 24, no. 5, pp. 1663-1682, 2023.

[17] P. S. Duth and S. G. Bhat, "Disease Classification in Citrus Leaf using Deep Learning," in *2022 IEEE International Conference on Data Science and Information System (ICDSIS)*, 2022, pp. 1-5: IEEE.

[18] V. Viswanatha, A. Ramachandra, P. T. Hegde, M. R. Reddy, V. Hegde, and V. Sabhahit, "Implementation of Smart Security System in Agriculture fields Using Embedded Machine Learning," in *2023 International Conference on Applied Intelligence and Sustainable Computing (ICAISC)*, 2023, pp. 1-6: IEEE.

[19] N. Schizas, A. Karras, C. Karras, and S. J. F. I. Sioutas, "TinyML for ultra-low power AI and large scale IoT deployments: A systematic review," vol. 14, no. 12, p. 363, 2022.

[20] M. El-Otmani, A. Ait-Oubahou, and L. Zacarías, "Citrus spp.: Orange, mandarin, tangerine, clementine, grapefruit, pomelo, lemon and lime," in *Postharvest biology and technology of tropical and subtropical fruits*: Elsevier, 2011, pp. 437-516e.

[21] M. Ismail and J. J. O. o. P. M. Zhang, "Post-harvest citrus diseases and their control," vol. 15, no. 1, p. 29, 2004.

[22] M. Mari, F. Neri, and P. J. S. P. R. Bertolini, "Novel approaches to prevent and control postharvest diseases of fruits," vol. 3, no. 6, pp. 1-7, 2007.

[23] S. Naqvi, "Diagnosis and management of certain important fungal diseases of citrus," in *Diseases of Fruits and Vegetables Volume I: Diagnosis and Management*: Springer, 2004, pp. 247-290.

[24] L. Mendonça, L. Zambolim, and J. J. J. B. M. O. A. Badel, "Bacterial citrus diseases: major threats and recent progress," vol. 5, pp. 4-2017, 2017.

[25] L. Timmer, S. Garnsey, and P. J. D. o. t. f. c. Broadbent, "Diseases of citrus," pp. 163-195, 2003.

[26] I. Talibi, H. Boubaker, E. Boudyach, and A. J. J. o. a. m. Ait Ben Aoumar, "Alternative methods for the control of postharvest citrus diseases," vol. 117, no. 1, pp. 1-17, 2014.

[27] A. U. Malik *et al.*, "Biotic and abiotic factors causing rind blemishes in citrus and management strategies to improve the cosmetic quality of fruits," 2021.

[28] N. Butt, M. M. Iqbal, I. Ahmad, H. Akbar, U. J. J. o. C. Khadam, and B. Informatics, "Citrus diseases detection using deep learning," vol. 6, no. 02, pp. 23-33, 2024.

[29] T. Poblete, A. Hornero, V. Gonzalez-Dugo, B. Landa, J. Navas-Cortes, and P. Zarco-Tejada, "Early Disease Detection with Hyperspectral Imagery: Dynamics of Plant Traits as a Function of Disease Severity Levels," in *IGARSS 2023-2023 IEEE International Geoscience and Remote Sensing Symposium*, 2023, pp. 2803-2806: IEEE.

[30] R. Arredondo Valdés *et al.*, "A review of techniques for detecting Huanglongbing (greening) in citrus," vol. 62, no. 10, pp. 803-811, 2016.

[31] J. Graham, T. Gottwald, T. Riley, and M. J. P. Bruce, "Susceptibility of citrus fruit to bacterial spot and citrus canker," vol. 82, no. 4, pp. 452-457, 1992.

[32] C. Roistacher, "Diagnosis and management of virus and virus like diseases of citrus," in *Diseases of Fruits and Vegetables Volume I: Diagnosis and Management*: Springer, 2004, pp. 109-189.

[33] S. Prakash *et al.*, "Is TinyML Sustainable?," vol. 66, no. 11, pp. 68-77, 2023.

[34] Y. Abadade, A. Temouden, H. Bamoumen, N. Benamar, Y. Chtouki, and A. S. J. I. A. Hafid, "A comprehensive survey on tinyml," 2023.

[35] S. Martínez-Fernández, X. Franch, and F. Durán, "Towards green AI-based software systems: an architecture-centric approach (GAISSA)," in *2023 49th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 2023, pp. 432-439: IEEE.

[36] S. Jain and R. Rajpoot, "Real-Time Monitoring and Disease Detection in Greenhouses Using IoT and Deep Learning," in *Computational Intelligence Techniques for 5G Enabled IoT Networks*: Springer, 2025, pp. 213-235.

[37] M. S. Murshed, C. Murphy, D. Hou, N. Khan, G. Ananthanarayanan, and F. J. A. C. S. Hussain, "Machine learning at the network edge: A survey," vol. 54, no. 8, pp. 1-37, 2021.

[38] S. Branco, A. G. Ferreira, and J. J. E. Cabral, "Machine learning in resource-scarce embedded systems, FPGAs, and end-devices: A survey," vol. 8, no. 11, p. 1289, 2019.

[39] D. Ashok, N. M. V. Nirmala, D. Srilatha, K. V. Rao, V. D. Babu, and S. J. Basha, "Leveraging CNN and LSTM for Identifying Citrus Leaf Disorders," in *2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS)*, 2023, pp. 730-735: IEEE.

[40] H. Kalim, A. Chug, and A. P. Singh, "Citrus leaf disease detection using hybrid cnn-rf model," in *2022 4th International Conference on Artificial Intelligence and Speech Technology (AIST)*, 2022, pp. 1-4: IEEE.

[41] E. G. d. Silva, F. d. S. Ponce, and S. G. H. d. J. B. J. o. B. Souza, "Technological advances and the use of IoT in monitoring Diaphorina citri in citrus cultivation," vol. 85, p. e290263, 2025.

[42] R. Injante and S. S. J. L. Isuiza, "Detection of citrus diseases using artificial intelligence: A systematic review," no. 3, p. 9, 2025.

[43] S. Zheng *et al.*, "Maturity recognition of citrus fruits by Yolov4 neural network," 2021, pp. 564-569 %@ 1665415401: IEEE.

[44] J. D. Ortiz Cuadros, O. Loyola-Valenzuela, and S. M. Rendón, "Grad-CAM Powered Transfer Learning for Citrus Disease Classification with MobileNetV2," in *Workshop on Engineering Applications*, 2025, pp. 356-367: Springer.

[45] R. Santhoshkumar, B. Rajalingam, P. Deepan, and P. Santosh Kumar Patra, "Smart Plant Leaf Disease Detection System using Internet of Thing (IOT) and PLDP Net-RF Model," in *Proceedings - 2022 4th International Conference on Advances in Computing, Communication Control and Networking, ICAC3N 2022*, 2022, pp. 476-482.

[46] X. Zhang, J. Zhang, L. Li, Y. Zhang, and G. J. S. Yang, "Monitoring citrus soil moisture and nutrients using an IoT based system," vol. 17, no. 3, p. 447, 2017.